2.0 CONFIGURATION MANAGEMENT BASELINE

This CM Baseline provides a general description of the model, its development history, its current status, and the change procedures and user support functions in place for TRAP. Although a procedural CM policy does not exist, the code appears to be well controlled by the model developer and model manager.

2.1 MODEL DESCRIPTION

TRAP is typically used to support the design and to evaluate the performance of threat air-launched weapons. It was designed specifically for modeling the performance of aerodynamic weapons, including AAMs, air-to-surface missiles (ASMs), and unmanned aerial vehicles (UAVs) against airborne or ground targets. The simulation and its intelligence-based threat model database reside at the National Air Intelligence Center (NAIC), Aerodynamic Weapons Design Branch, and is provided to many tri-Service users and their contractors. It is also distributed as part of the Survivability/Vulnerability Information Analysis Center (SURVIAC) aircraft survivability family of codes.

TRAP can simulate up to three vehicles: a launch aircraft, a missile, and a target aircraft. Although often referred to as an "engagement" model, TRAP does not strictly model an engagement. Only the launch aircraft carries a weapon and, although ground and airborne targets are modeled, this is only to the extent that they provide a target to which a missile can be guided. The modeling emphasis is on the missile fly-out portion of the engagement.

Four kinematic missile models are implemented: modified point-mass, 3-DOF, 5-DOF, and 6-DOF. The modified point-mass model uses information about the airframe, mass properties, and propulsion to calculate the instantaneous angle-of-attack (AOA) and angle-of-sideslip (AOS) needed to pull the required accelerations for intercept, subject to a maximum rate of change specified by the user that reflects missile agility. This allows synthetic body rates to be generated, which allows a detailed seeker model to be coupled to a point-mass airframe model. The N-DOF models are built on the modified point-mass digital model. For these more detailed simulations, the missile guidance and control (G&C) system components are modeled in whatever detail is appropriate or possible, and output fin deflections are passed to the aerodynamics for the computation of the forces and moments on the missile body.

TRAP can be used to simulate single-shot launch scenarios, multiple stacked shots, or in an iterative mode to generate the maximum and minimum launch acceptability region (LAR) boundaries in both the horizontal and vertical planes. The simulation can also be used in a missile performance reconstruction capability, to estimate unknown missile attributes given a set of known data and observations. It does not model fuzing or warhead effects.

The following is a summary of TRAP V.3.1a capabilities:

- Missile Seeker: active, semi-active, or infrared (IR) with limited noise modeling and signal processing. Generates radio frequency (RF) and IR signal-to-noise ratio.

- Missile Guidance Types: proportional navigation (pro-nav) acceleration or rate, pursuit, biased or deviated pursuit, constant altitude, constant gamma, constant g-maneuver, three-point, lead angle, pre-programmed acceleration, altitude, cross-range.
- Missile Control Types: canard, tail, wing, wing-tail, thrust vector.
- Missile Autopilot Types: acceleration feedback with rate damping, midship and forward accelerometer feedback, synthetic stability control, body rate control, body attitude control, torque balance control, fin-position control, alpha-beta control, thrust vector control (TVC).
- Missile Aerodynamics: modified point-mass (drag, lift, and angle of attack) through full longitudinal and lateral force and moment calculations for up to 6-DOF. All aerodynamic coefficients generated with MISDAT (MISsile DATcom) aero prediction method.
- Missile Propulsion: constant or varying solid or liquid rocket, air-breather, or any combination, or none. Full throttle capability.
- Missile Mass Properties: single or multiple stage, linear or other mass burn-off, moments-of-inertia input for 3- to 6-DOF simulations.
- Equations of Motion: right-hand-rule coordinate system with Z-positive down. Equations of motion from Etkin.
- Integration: Euler / Tustin coefficients.
- Missile Launch Envelope Generator: grid or semi-smart (automated boundary search) methods against non-maneuvering or maneuvering target. Auxiliary polar plot capability for launch envelope presentation.
- Endgame: termination at point of closest approach; kill recorded if point-of-closest-approach (PCA) within input lethal radius.
- Missile Performance Reconstruction: mathematical search and convergence technique (complex box search with weighted least squares) to estimate unknown data attributes, such as bias and scale factor, given known data and observations as constraints.
- Launch Aircraft: simplified (straight and level) or modified point-mass system (AOA and simplified pitch and roll dynamics). Launch aircraft may use pursuit algorithms or perform generic maneuvers. For simplified point-mass applications, there is no aerodynamic or engine input; for modified point-mass, aircraft maneuvers are constrained by its aerodynamic and propulsion capabilities.
- Target Aircraft: simplified or modified point-mass system similar to launch platform. This vehicle can be stationary (e.g., ground target) or maneuverable with constant velocity or acceleration in any plane. The modified point-mass option allows the target to maneuver according to a user-choice of pre-programmed or dynamic 'generic' maneuvers; target will maneuver constrained by its own aerodynamic and propulsive capabilities. For the simplified point-mass simulations, no aerodynamics or engine model is input and the target does not maneuver.

- Second Target: Follows a flight-path that is offset from that of the primary target (the relative positions of the two targets is determined by the type of maneuver being executed).

Input. A TRAP model consists of single parameter data files, multiple parameter table data, and any customized FORTRAN source code specific to the missile. These files include inputs for missile characteristics, as well as a scenario control file and any files associated with a non-generic launch or target aircraft. TRAP is designed to be a data-driven simulation. A user-driven top level FORTRAN interface to the TRAP main body of subroutines allows the user to schedule key characteristics that are important for a particular missile's flight. It also provides an interface to other user-written routines, if necessary, to accurately model the aerodynamic vehicles and their subsystems.

A digital model database in TRAP is maintained and distributed by NAIC, and consists of 87 current aerodynamic weapon systems, comprehensively covering foreign air-to-air and air-to-surface threats. These digital models of threat weapon systems are used as an input to TRAP. All TRAP digital models and data files have individual classifications, with the highest classification being SECRET/NOFORN. Table 2-1 lists the number of threat models available by type and nationality as of 1 June 1995. A brief synopsis of each threat model is given in the NAIC/TANW Export Summary Memo [11], which is published semi-annually.

TABLE 2-1. Threat Models in TRAP V.3.1a.

Vehicle	Nationality	Number
Air-to-Air Missiles	Red, Grey	54
Air-to-Air Missiles	Blue	3
Air-to-Surface Missiles	Red, Grey	26
Rockets	Red	1
Bombs	Red	1
UAVs	Red, Grey	2

Threat Model Datasets Include:

- Single Parameter Data File:
  - missile aerodynamics (aero), autopilot, and guidance parameters
  - missile mass properties
  - missile propulsion properties
  - missile seeker parameters
  - user-defined variables
  - scenario set-up
  - output control (instrument panel)
  - plot control

- Table Data Files
  - complete aerodynamic look-up table
  - missile moments-of-inertia (not applicable for point-mass simulations)
  - gain scheduling with respect to dynamic pressure
  - complete thrust profile table

- Any other FORTRAN files unique to a system
- POLICY (top-level subroutine)

Output. TRAP produces a time-history of any variable within the simulation, as specified by the user, to aid in analyzing the performance of the missile during a flight. TRAP also creates output files summarizing the results of LAR searches, including F-Pole and A-Pole results when applicable. TRAP does not model end-game effects.

Host Platforms, Systems, and Software. TRAP is hosted at NAIC on SUN SPARC stations and on an IBM Mainframe computer, using a FORTRAN Version 2 Compiler. It has been hosted by users on IBM, SUN, VAX, SGI, IBM PC, Power Macintosh, DEC, Motorola, and HP computers. A summary of present user host platforms and compilers is contained in Section 4.

Capabilities. The TRAP simulation, its associated classified missile datasets, and the results these models generate, provide the official Air Force performance assessment for many foreign air-launched weapon systems. Additionally, TRAP models are used as inputs to larger, more aggregate air-to-air combat simulations, and as the fly-out model for many Hardware-in-the-Loop (HITL) and dome simulators. A summary of overall simulation capabilities is provided in Table 2-2.

TABLE 2-2. Summary of TRAP Capabilities.

Title	**TRAP - TRajectory Analysis Program**
Model Type	Engineering Analysis
Proponent	National Air Intelligence Center, WPAFB, OH
Point of Contact	Mr. Joseph Herrmann, NAIC/TANW (Model Developer)
Purpose	Air-Launched Weapons Flyout
Domain	Air-to-Air, Air-to-Ground
Span	Individual
Environment	Atmospheric properties for current altitude of target(s) and launch aircraft determined for aerodynamic and propulsion modeling. Clutter and multipath models used for ground-based target. For missile radar seekers, options exist for programming multipath, clutter, and jamming signals. For IR seekers, code to calculate atmospheric transmittance not yet operational.
Force Composition	Launch Aircraft, Missile, Target (single engagement)
Scope of Conflict	Short-Range to Long-Range Air-to-Air, Air-to-Ground
Mission Area	Aircraft Survivability, 1 v. 1, Air-to-Air Missile Engagement, Air-to-Surface Engagement
Level of Detail of Processes and Entities	Missile Model from Point-Mass to 6-DOF Aircraft Model Simplified or Modified Point-Mass
Human Participation	Not Required
Time Processing	Dynamic
Sidedness	One-Sided
Limitations	1 v. 1
Planned Improvements and Modifications	Version 4.0. Quaternions, Graphical User Interface, Missile Aero Method Improvements
Input	Scenario, Single Value & Tabular Data Files

TABLE 2-2. Summary of TRAP Capabilities. (Contd.)

Title	TRAP - TRajectory Analysis Program
Output	Depends on simulation type and user selection. An event-by-event tabular file is generated, as are files for tabulation and plotting of simulation results. The user can also select generation of a file for 3-dimensional graphics display.
Computers (operating systems)	IBM PC, MICROVAX, VAX, SGI, SUN, IBM, DEC, MOTOROLA, HP, POWER MAC, (UNIX, MS-DOS, VMS*, VMS-CMS*)
Storage	7 Mb
Peripherals	Graphics
Language	FORTRAN 77
Documentation	Software User's Manual; Design Engineering Memos
Security Classification	Core Simulation is Unclassified Models are Confidential - Secret/Noforn
Date Implemented	1981
Data Base	Model-Specific Database supplied by NAIC
CPU time per cycle	Variable
Data Output Analysis	Available
Frequency of Use	User-Dependent
Users	Approximately 144
Comments	Accepted by DOT&E as Baseline Simulation for use in real-time Air-to-Air applications for Electronic Combat (EC) Test Communities and on Open Air Ranges (OARs)
* Virtual Machine System-Conversational Monitor System	

2.2 DEVELOPMENT HISTORY

In 1978, the Air Force Foreign Technology Division (FTD) perceived a need for a trajectory analysis tool. In 1979, a development task was contracted to Applications Research Corporation (ARC) of Dayton, Ohio, which delivered the Air-to-Air Missile Evaluation Program (AAMEP) in 1981. Although AAMEP had a library-like structure of autopilot, guidance types, and propulsion, with a versatile set of options, its structure was deemed by FTD to require improvement. FTD restructured the program, rewrote most of the code, developed a user-friendly input scheme, and renamed the code TRAP. This model became an in-house analytical tool for evaluating flight trajectories of threat air-to-air missiles.

<u>Version 2.0</u>. Additional capabilities were added in Version 2.0 and the role of TRAP was expanded from the evaluation of AAM flight trajectories to the evaluation and prediction of the design and performance of missiles in one-on-one engagements. Version 2.0 was composed of approximately 13,000 lines of FORTRAN 77 source code in 155 subroutines and 28 common blocks. The program required approximately 220K of memory. The Institute of Mathematics Statistical Library (IMSL) was used for random number generation. [12]

Wide dissemination of the program provided a vehicle for FTD release of a working system model. As other organizations examined and used TRAP, more was learned about modeling missile aerodynamics at the engagement level. Contractors integrated subsystem models and TRAP was in a fairly constant state of upgrade and improvement, both to enhance the program's technical capability and to increase its utility as a performance analysis tool for FTD customers.

Version 3.0. In April 1989, the program was placed in the SURVIAC library. Because the release of a new version was imminent, distribution from SURVIAC was delayed until the release of V.3.0 in January 1990. Version 3.0 consisted of approximately 20,000 lines of FORTRAN 77 source code in 256 subroutines and nine common blocks. The program required about 3 megabytes of memory. The IMSL was used for random number generation. [13]

Version 3.1. The next version was completed in September 1991, and placed in the SURVIAC library in January 1992. Version 3.1 consisted of approximately 30,000 lines of FORTRAN 77 source code in over 300 subroutines and 91 common blocks. The program required about 3 megabytes of memory. TRAP 3.1 is currently being run in a number of user applications. [6]

Personal Computer TRAP (PCTRAP). In addition to the versions listed above, a streamlined, quick-look version of TRAP, called PCTRAP, is also distributed by NAIC. PCTRAP is used in the operational community as a briefing/debriefing tool, such as by Fighter Weapon Schools, and for test mission analysis, such as by the 57th Test Group at Nellis AFB. This version is also used to a lesser degree by the DIA and military-industrial complex.

The PCTRAP code is smaller in size and faster in execution time than TRAP, which was accomplished by simplifying the simulation in all areas. Each of the 40+ missiles presently contained in PCTRAP are based on original TRAP models. This document does not address PCTRAP other than to mention that this capability exists.

2.3 VERSION DESCRIPTION AND CURRENT STATUS

TRAP Version 3.1a was placed in the SURVIAC library in July 1993. The current version consists of approximately 36,000 lines of FORTRAN 77 source code in over 340 subroutines and 91 common blocks. The program requires about 4 megabytes of memory and the IMSL is used for random number generation.

Major changes from the previous version include:

- The addition of generic aircraft maneuver routines
- More detailed or dynamic modeling that changes as a function of intercept geometry, accomplished through the user-written POLICY routine or other user-written FORTRAN routines
- Corrections to sections of the simulation that contained inconsistencies or errors
- Removal of a statistical error analysis package

Configurable Items. Version 3.1a is the current main body of routines. Applicable configuration items are shown in Table 2-3. The individual model databases, as previously listed, are provided to the user along with, in some cases, a technical bulletin discussing the model and performance data.

TABLE 2-3. Configuration Items for TRAP V.3.1a.

Description	Classification	Name/ Document #	Date
Software Source Code, Magnetic Media	Unclassified	TRAP 3.1a	July 2, 1993
Software User's Manual (SUM)	Unclassified	TAG No. 91-02D TAG No. 92-03	March 31, 1993
Errata Sheets	Unclassified	TRAP 3.1a Errata Sheet #1	July 2, 1993
Digital Models (87)	Confidential-S/NF/W*	Various	Various
Engineering Design Memos	Classified	Various	Various
(*) WNINTEL classification deleted by DCI Directive 1/7, 12 Apr 95			

The TRAP V.3.1 SUM listed in Table 2-5 is produced by Battelle of Dayton, Ohio, under a contract with NAIC. It is distributed by SURVIAC, as requested by the user. The set of seven User Manuals does not include a Software Analyst's Manual (SAM) or Software Programmer's Manual (SPM); however, the SUM contains much of the information typically included in these documents.

Current program size totals are summarized in Table 2-4.

TABLE 2-4. TRAP V.3.1a Program Size Totals.

Number of Files on Distribution Media	466
Number of Main Programs	1
Number of Subroutines	350+
Number of Functions Imbedded	12
Number of Functions (as separate sub-routines)	13
Number of Common Blocks	91
Total Source Lines Of Code	36,000
Total Size (Kilobytes)	4 Mb
Total Size of Compiled Executable Code	~ 2 Mb

2.4 CHANGE PROCEDURES

TRAP was designed as an in-house tool, a general purpose simulation created by the intelligence community for their own use. As a result, in-house CM procedures were created to satisfy management and control requirements. The development process is accomplished in an ad hoc manner. Although no procedural CM policy exists, the code appears to be well-controlled by the model developer and manager.

The model manager generates an informal development plan based on intelligence community inputs, problem/suggestion reports, and funding limitations. The TRAP Development Council (TDC) consists of eight NAIC engineers and functions in lieu of a formal Configuration Control Board (CCB). The TDC meets monthly to review the development plan and prioritize changes to the model based on the perceived need by the intelligence community. Requests for modifications from outside the intelligence community are also addressed to the extent to which they are understood. There appears to be no good conduit for transfer of information between NAIC and the ‘outside’ user community. When joint actions are required, the Branch Chief acts as a configuration manager for the TDC. The TDC composition is shown in Table 2-5.

TABLE 2-5. TRAP TDC Composition.

Function	POC	Office	Phone
TDC Chair Branch Chief, Aero Engineer	Doug Samuels	NAIC/ TANW	DSN 787-2653 C: 513-257-2653 FAX: Ext 9888
Model Manager (/Developer), Sr. Branch Analyst, Aero Engineer	Joe Herrmann	"	"
Sr. Branch Aero Engineer	Betsy Witt	"	"
Sr. Branch Analyst IR Seekers, 6-DOF	Carlos Colon	"	"
Sr. Branch Analyst UAVs	Craig Logan	"	"
Sr. Branch Analyst Airborne Intercept (AI), RF, & Anti-Radiation Missile (ARM) Seekers	Rob Luczka	"	"
Sr. Branch Analyst Air-Air (A-A) Propulsion	Don Davis	"	"
Sr. Aero Engineer Air Surface Missiles (ASM)	Bill Boyee	"	"
Jr. Engineer Aircraft Modeling, Documentation	Ed Jenkins	"	"
Technician & Administration Export Models	Bonnie Whitescarver	"	"

When modifications are deemed necessary, engineers from NAIC or Battelle are assigned as appropriate to implement those changes. Upon completion, the updated code module is given to the Model Manager (MM) for integration and regression testing, which constitutes in-house V&V at NAIC. Interim changes are documented in Errata Sheets. The MM then forwards the change release to SURVIAC for distribution. Figure 2-1 depicts the TDC action flow.

There are no documented guidelines or procedures for the conduct of the TDC. There does not appear to be any future plans to replace the current TDC with a formal CCB. NAIC views TRAP as a threat analysis support tool, and presently there is no requirement, obligation, or funding to respond to other than in-house and intelligence community configuration management problems. The entire TRAP change, development, and maintenance process is shown in Figure 2-2.

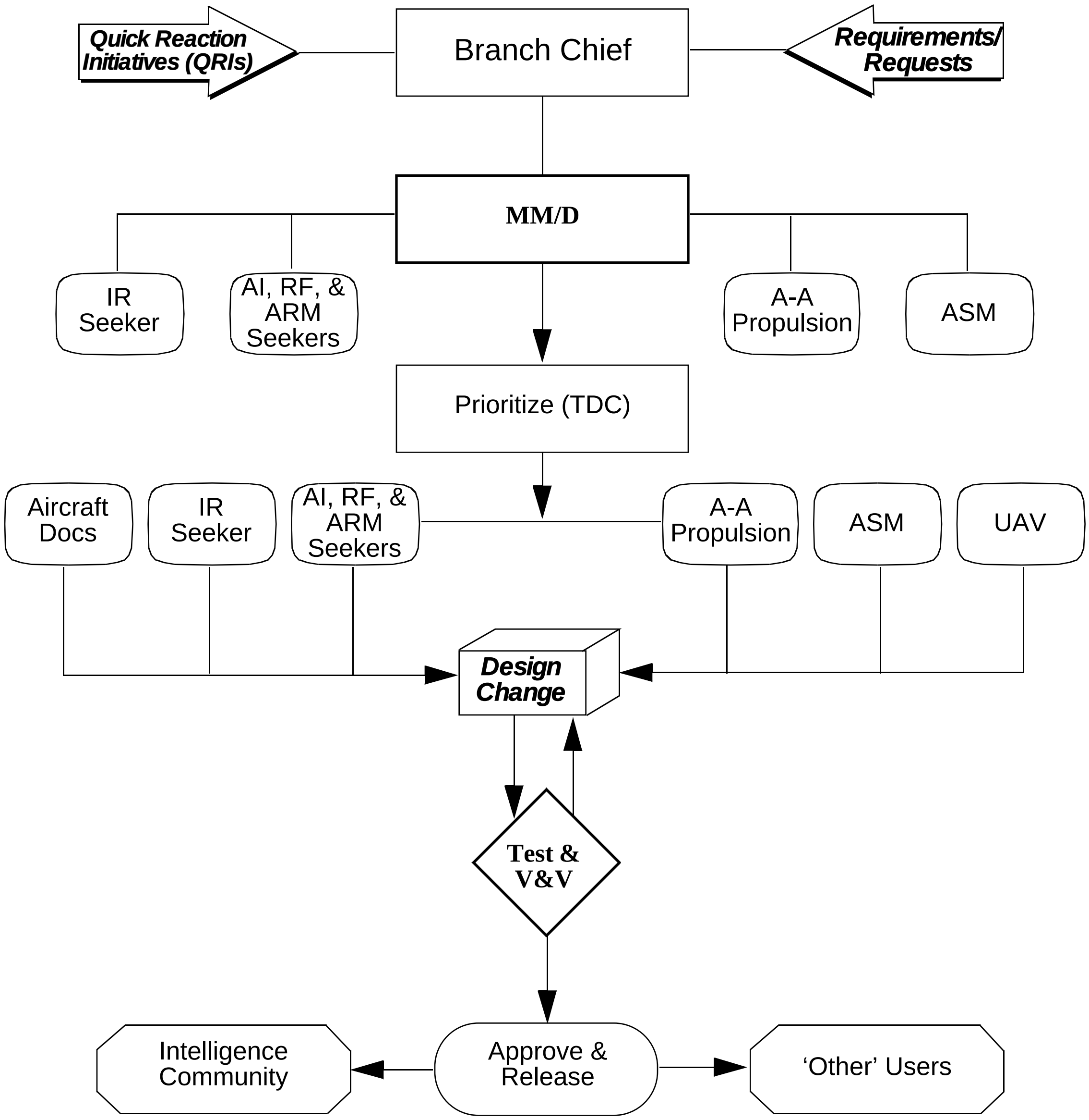

FIGURE 2-1. TDC Action Flow.

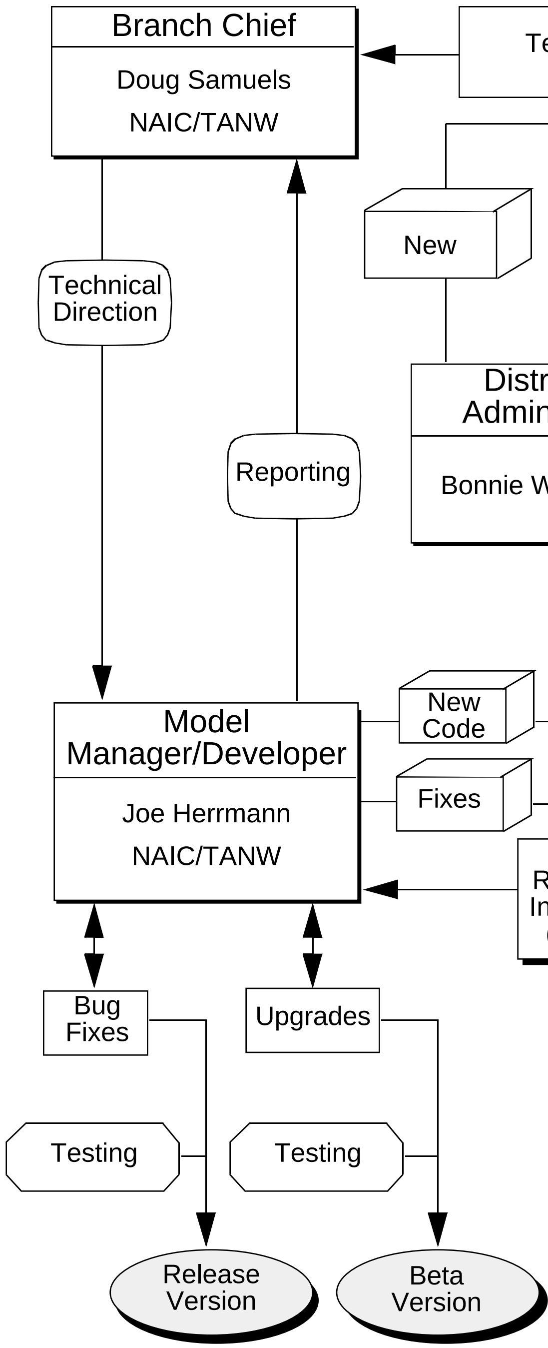

FIGURE 2-2. TRAP Change, Development, and Maintenance Flow.

SURVIAC notifies users through their newsletter and distributes the code as requested to government agencies and government sponsored contractors. Occasionally, NAIC will send a pre-release version of the code to special government users who have an immediate need for the model.

2.4.1 Change and Version Identification Policy

While the model has progressed from TRAP to Version 2.0 to Version 3.1a, there is no documented version identification policy. Approximately 2 years elapsed between the release of 3.0 and 3.1, a year and a half between 3.1 and 3.1a, and almost 3 years between the release of 3.1a and the beta version of 4.0. As of October 1997, Version 4.0 has not been entered into SURVIAC.

Model Deficiency Reports (MDRs)

There is no formal MDR process for TRAP. It is recommended that users contact the model manager directly with problems or requests for changes.

2.4.2 Beta Site Development Policy

New versions of the model and associated documentation are sent to approximately eight beta sites for test and evaluation. The code and documentation are updated to correct any problems found during beta test and then delivered to SURVIAC for distribution. NAIC has proposed an initiative to establish four alpha sites. The alpha sites would evaluate the code and documentation and the beta sites would test and evaluate overall simulation performance.

2.5 USER SUPPORT FUNCTIONS

Although no formal MDR process exists for TRAP, NAIC has identified the lack of feedback from ‘outside’ users as a major roadblock to model improvement. Users conferences provide the only open forum for discussion of upgrades, corrections, deficiencies, and user requirements. Conferences were hosted on 17 November 1989, 27 June 1990, 16 October 1991, and 19-20 May 1994, but have been held too infrequently to facilitate communications among all users. This is due, in part, to the fact that there are no personnel dedicated solely to TRAP, and the resources to fully support outside users is not available. TRAP was designed to be an in-house tool; thus, in-house requirements for enhancements or corrections take precedence over outside users’ requirements.

SURVIAC serves as the distribution manager of TRAP in that they ensure that registered users have access to the most recent version of the software and documentation. The software and documentation must be obtained through SURVIAC with separate line item requests. TRAP users are not notified immediately when a new version is released; it is only through the quarterly SURVIAC bulletin, which lists when updates are available, or by word of mouth, that users become aware of a new model/code/documentation version. NAIC handles all requests for threat model data sets.

Users typically rely on direct contact with NAIC for support. In a survey distributed to the user community, users were asked who they relied on for configuration management of the model and what standards and practices they used for CM. Of the 41 users who responded, 15 had no practices or standards, 14 relied on in-house standards, 5 relied on SURVIAC configuration management, and four relied on User Group Meetings.

SURVIAC serves as the model distributor rather than the configuration management agent for TRAP. SURVIAC itself does not maintain a TRAP simulation support capability; only sample cases can be run. User-generated problem/suggestion reports on the distributed code are sent to NAIC. Requests by users to SURVIAC for technical assistance are normally referred to NAIC/TANW. Given the appropriate resources, SURVIAC could fulfill its role as “technical expert” and “customer service agent” for the model, and provide a channel for information flow between NAIC and the user community.

2.6 IMPLICATIONS FOR MODEL USE

TRAP is maintained by the NAIC, Aerodynamic Weapons Design Branch. Although no procedural CM policy exists, the code appears to be well controlled by the model manager and the Trap Development Council.

The TDC prioritizes all requests for change based on the perceived need by NAIC and the intelligence community. The TDC decides whether or not a particular modification should be made, and if so, how the change should be implemented. Interim changes are documented in Errata Sheets. Fixes spurred by internal V&V efforts at NAIC are not normally tracked or documented.

Master copies of the released code version and any associated documentation are maintained at SURVIAC. Distribution of new releases is not automatic and users are not notified when new software and documentation versions are released.